

ソフトウェア開発支援環境におけるプロセス記述言語仕様に対する一考察

市原 利浩, 三浦 昭浩[†]

三菱電機(株) 情報技術総合研究所[‡]

1. はじめに

ソフトウェア開発プロジェクトの大規模化に伴う生産性の低下や品質問題に対応する手段として、CMMI[1]による開発プロセス改善活動が注目を集めている。しかし、大規模プロジェクトでの開発プロセスの改善は時間を要する活動である。この改善を効率的に進めるべく我々は、ソフトウェア開発支援環境(以下、プロセスナビゲータ)の開発を進めている。プロセスナビゲータは、ソフトウェアの開発手順の定義をプロセス記述として与え、この定義を基に作業手順を示し、適切なタイミングでツール起動を行い、開発者の作業を誘導する。

本稿では、プロセスナビゲータで利用するプロセス記述言語について、言語に対する要求と、それに対する制御機構について議論する。

2. ソフトウェア開発支援環境

2.1. プロセスナビゲータにおける課題

プロセスナビゲータは、事前に定義したプロセス記述に従って動作し、開発者に作業手順の提示やツール起動等を Web 環境上で提供することにより以下のことを狙っている。

- プロセス改善活動によって改訂した標準開発プロセスの組織内での普及・定着
- 品質データの計測・分析によるプロダクト品質の管理
- 各作業に対して、適切なツールの自動起動や、参照データの提示等による生産性の向上
- 組織内での情報共有、及び開発者間の確実なコミュニケーション支援

これらの実現手段として、固定的な手順に基づく従来のワークフローを用いることが考えられるが、実際のソフトウェア開発では以下の点に問題がある。

- 組織標準として定義された標準開発プロセスは、プロジェクトごとに開発規模や期間などの特性が異なるため、そのまま全てのプロジェクトに適用するのは困難である。
- 開発手順は進捗状況などにより変更することが多く、事前に全ての手順を予測してプロセスを定義することは難しい。従来のワークフローにおけるプロセス記述は、事前に全ての作業手順を定義するというスタイルをとっているため、実作業とナビゲーションする作業手順の乖離が大きくなってしまう場合がある。

2.2. プロセス記述言語の課題

2.1 節の解決策として、以下の機能を実現する。

- プロジェクトの規模、開発期間などの特性(以下プロファイル情報)をもとにした標準開発プロセスのテーラリング
- プロセスの作業状況に応じた動的なプロセスの生成・変更

この機能を実現するためには、現状のプロセス記述言語の制御機構を拡張する必要があり、その際の課題を以下に示す。

プロファイル情報に基づき、柔軟に標準開発プロセスから実際にプロジェクトに適用する実行プロセスを定義可能とする仕組み

プロセスを動的に組み合わせる際、矛盾が発生しない適切なライブラリを選択して、実行プロセスを生成できる仕組み

動的変更したプロセスの実績を評価し、ライブラリの選択条件を見直す仕組み

Process Definition Language for Software Process Navigation

[†] Toshihiro Ichihara, Akihiro Miura

[‡] Information Technology R&D Center, Mitsubishi Electric Corporation

3. 言語に対する要求と制御機構

2章で述べた課題に対して、検討したプロセス記述仕様の方針と制御機構を以下に示す。

プロセステーラリング機能

テーラリング機能では、過去の類似プロファイル情報を有するプロジェクトにおいて、標準開発プロセスから逸脱した事例が存在した箇所などを抽出、プロセスの動的生成・変更箇所（以下、変更ポイント）を実行プロセス内に設定する。また、プロセスの動的生成・変更を実施する場合にライブラリを選択する選択条件をプロファイル情報に基づき、設定する。

プロセス動的生成・変更機能

プロセスの動的生成・変更機能では、開発者の作業が事前に実行プロセス内の変更ポイントに到達すると、実績値を基に組合せ可能なライブラリ群から予め設定された選択条件をもとにライブラリを選択、動的に実行プロセスに組込む。また、プロセスの動的生成・変更時に矛盾が発生しない組合せ可能なライブラリを抽出するために、プロセス記述に機能を分類するカテゴリや入出力情報を定義する。

実行プロセス評価機能

プロジェクト終了後に、不具合発生件数や作業時間などの実績値を計測し、統計処理する。統計処理した値からライブラリの後工程に与える影響度（成果物に対する品質，開発期間）を評価する。その結果をもとにプロセスの動的生成・変更時におけるライブラリの選択条件を見直す。

4. まとめ

今後、今回検討したプロセスのテーラリング及びプロセスの動的生成・変更機能を、現状のプロセスナビゲータに実装させる予定である。なお、プロセス記述言語の記述の容易性や可読性を向上させるエディタ、可視化ツールなどの開発が課題として残っている。今後は、開発現場と協力して、プロセスナビゲータの運用と機能評価を進めると共に、プロセス記述の洗練化の検討を進めていく予定である。

参考文献

- [1] M.C.Paulk,et.al.,, *Capability Maturity Model, The:Guidelines for Improving the Software Process*, Addison-Wesley Publishing(1995)
- [2] 井上克郎 他、ソフトウェアプロセス、共立出版（2000）

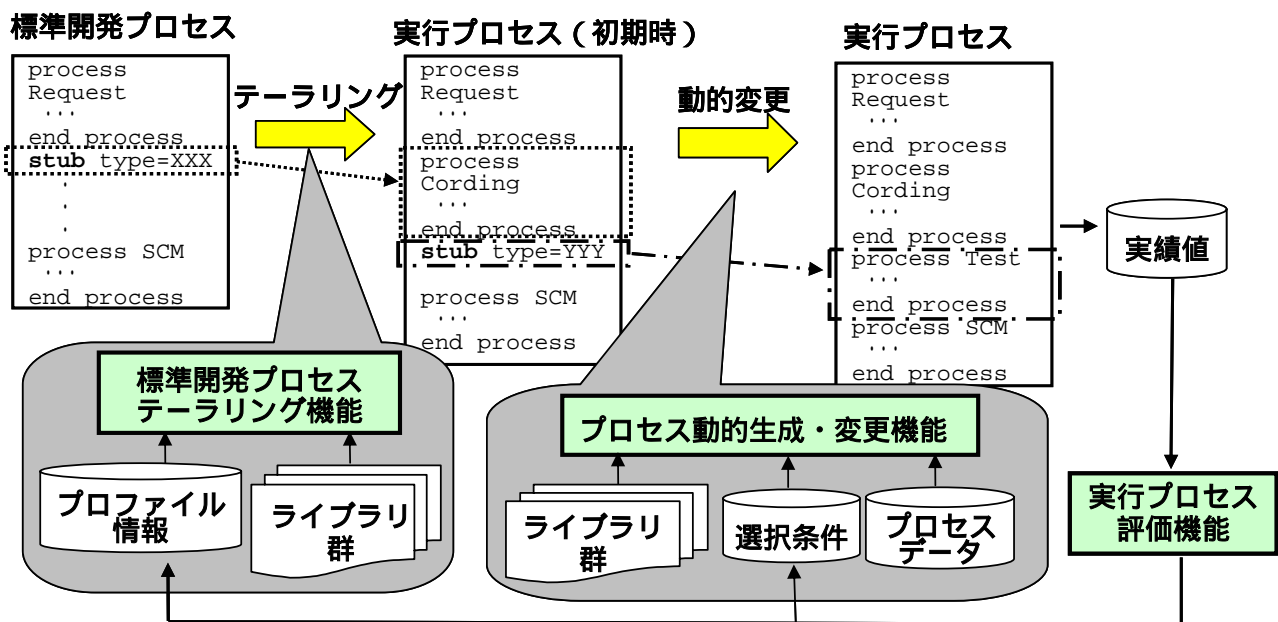


図 1 プロセスのテーラリングと動的変更